



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/607,397	06/30/2000	Damon Barry	13768.132	9886
47973	7590	10/06/2005		
WORKMAN NYDEGGER/MICROSOFT 1000 EAGLE GATE TOWER 60 EAST SOUTH TEMPLE SALT LAKE CITY, UT 84111			EXAMINER KISS, ERIC B	
			ART UNIT	PAPER NUMBER
			2192	

DATE MAILED: 10/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/607,397

Applicant(s)

BARRY ET AL.

Examiner

Eric B. Kiss

Art Unit

2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 13 July 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-4, 7, 10-17 and 20-27 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-4, 7, 10-17, and 20-27 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

2

DETAILED ACTION

1. The reply filed 13 July 2005 has been received and entered. Claims 1-4, 7, 10-17, and 20-27 are pending.

Response to Arguments

2. Applicant's arguments filed 13 July 2005 have been fully considered but they are not persuasive.

In response to Applicant's arguments, the Examiner maintains that the test suites of TETware are made up of one or more test cases, the filename of the test cases correspond to one or more test suites (*i.e.*, they are specified relative to the test suite directory), and the scenario file indicates which of the one or more available test cases of a particular test suite are selected to be executed (see, for example, section 5.3.2 of TET_UG; more specifically, see 5.3.2.4, describing how test case filenames are processed).

The Examiner further maintains that a scenario file contains one or more scenarios and scenarios contain test cases (see section 4.1 of TET_PG and 5.3.2.1 of TET_UG). When an individual scenario from the scenario file is processed, one or more test cases may be invoked (as described, for example, 4.2.4.3 of TET_PG and 5.3.2.4 of TET_UG).

Claim Rejections - 35 USC § 102

3. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

4. Claims 1, 2, 4, and 7-27 are rejected under 35 U.S.C. 102(b) as being anticipated by the TETware Release 3.3 software product (hereinafter TETware) released September 18, 1998 by The Open Group, as evidenced by: “TETware User Guide, Revision 1.2” (hereinafter TET_UG), “Release Notes for TETware Release 3.3” (hereinafter TET_RN), and “TETware Programmers Guide, Revision 1.2” (hereinafter TET_PG).

As per claim 1, TETware is disclosed with a computer system comprising:
one or more program modules (test suite) storing one or more available test cases (see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed), each comprising a set of instructions for testing a feature of the computer program through a language and format independent interface (the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG; the use of different source languages to build cases is also disclosed, e.g., C, C++, Shell, Korn Shell, or Perl; see, for example, section 2.4 of TET_UG describing the API components as linkable object code);

a harness client comprising a set of instructions that (i) receives user input specifying one or more filenames corresponding to the one or more program modules (see, for example, section 5.3.2 of TET_UG), (ii) employs the connector to scan for and discover the one or more available test cases that are stored in the one or more program modules and to organize the one or more available test cases into a test case hierarchy (see, for example, section 5.3.2 of TET_UG), and

Art Unit: 2192

(iii) receives user input for indicating which of the one or more available test cases in the test case hierarchy are selected to be executed on the computer program (see, for example, section 5.3.2.2 of TET_UG);

a harness comprising a set of instructions that (i) receives the test case hierarchy, (ii) traverses the test case hierarchy, and (iii) executes each of the one or more available test cases that is selected to be executed (test scenario) on the computer program using the corresponding language and format independent interface of the selected test case to ensure that the computer program processes as intended (test case controller; see sections 2.1 and 2.2 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG);

a connector, comprising a set of instructions that (i) scans for the one or more available test cases stored in the one or more program modules, (ii) organizes the one or more available test cases into the test case hierarchy by extracting the one or more available test cases from the one or more program modules (see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed; section 2.4 of TET_UG; and section 2.4.4 of TET_PG describing the handling of non-API test cases), and (iii) selectively integrates an interface between the test case hierarchy and the harness regardless of the language or format in which the one or more available test cases were written (test case managers and API libraries; see section 2.4 of TET_UG; see also section 2.4.4 of TET_PG describing the handling of non-API test cases);

a processor for executing each selected test case, the harness, the harness client, and the connector (inherent in the operation of the UNIX and WINDOWS operating systems used to implement TETware; see section 1.1 of TET_UG).

TETware is further disclosed with one or more test cases comprising a test suite in the hierarchy and one or more test suites comprising a test module in the hierarchy (see section 2.2 of TET_UG; see further, section 4.1 of TET_PG and 5.3.2.1 of TET_UG). When an individual scenario from the scenario file is processed, one or more test cases may be invoked (as described, for example, 4.2.4.3 of TET_PG and 5.3.2.4 of TET_UG).

As per claim 2, TETware is further disclosed with the set of instructions of the harness and the set of instructions of the connector utilizing an architecture that defines a means for accessing a resource over a network (see section 2.6.3 of TET_UG).

As per claim 4, TETware is disclosed with a method comprising:

The harness client receiving user input that (i) specifies a search property to identify one or more test cases of interest (see, for example, section 5.3.2 of TET_UG), (ii) selects one or more test cases from the one or more test cases of interest to execute on the computer program (see, for example, section 5.3.2.4 of TET_UG), and (iii) specifies how the one or more selected test cases are to be executed on the computer program (see, for example, section 5.3.2.2 of TET_UG);

the connector scanning the binary program module (test suite) storing the plurality of individually accessible test cases, for one or more test cases of interest (see, for example, section

Art Unit: 2192

2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed), each test case having a language and format independent interface for executing the test case on the computer program regardless of the language or format used to develop the test case (the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG);

the connector extracting the one or more test cases of interest from the binary program module (see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed);

the connector organizing one or more test cases into a test case hierarchy (test suite structure; see section 2.2 of TET_UG; see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed);

the connector interfacing a harness with the one or more test cases of interest (see section 6.4 of TET_UG; see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed), wherein the interfacing allows the harness to recognize and execute the one or more test cases of interest regardless of the language or format in which the one or more test cases of interest were developed (test case controller; see sections 2.1 and 2.2 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG); and

the harness traversing the test case hierarchy and executing each of the one or more selected test cases to test the computer program (see the description of the test case controller beginning on page 105 of TET_UG).

TETware is further disclosed with one or more test cases comprising a test suite in the hierarchy and one or more test suites comprising a test module in the hierarchy (see section 2.2 of TET_UG; see further, section 4.1 of TET_PG and 5.3.2.1 of TET_UG). When an individual scenario from the scenario file is processed, one or more test cases may be invoked (as described, for example, 4.2.4.3 of TET_PG and 5.3.2.4 of TET_UG).

As per claim 7, TETware is further disclosed with a step of determining whether one or more of the test cases of interest are identified as being deselected, wherein a deselected test case is not executed on the computer program (see, for example, the “-n” command line option of the test case controller on page 107 of TET_UG).

As per claims 10 and 11, TETware is further disclosed with excluding test cases determined to be deselected from a selection of a test suite or scenario (see, for example, the “-n” command line option of the test case controller on page 107 of TET_UG).

As per claims 12-14, TETware is further disclosed with the step of traversing further including executing the one or more test cases on a thread pool comprising one or more threads, and further discloses testing single-threaded and multi-threaded (thread-safe) models (see section 17.4 of TET_PG).

Art Unit: 2192

As per claims 15-17, these are computer-readable medium versions of the method discussed above (claim 4), wherein all limitations have been addressed as set forth above. Furthermore, the use of such a computer-readable medium containing executable code is inherently necessary for the operation of the UNIX and WINDOWS operating systems used to implement TETware (see section 1.1 of TET_UG).

As per claim 20, TETware is further disclosed with user-selected (through a harness client user interface) test cases (see the description of the test case controller and command line usage beginning on page 107 of TET_UG).

As per claims 21-23, see the disclosure applied above in the rejection of claims 12-14.

As per claim 24, TETware is disclosed with a method comprising:

specifying one or more filenames for identifying one or more program modules storing one or more test cases, each comprising a set of instructions for testing a feature of the computer program through a language and format independent interface (see, for example, section 5.3.2 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG);

identifying the one or more test cases within the one or more program modules (see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed);

Art Unit: 2192

translating the identified one or more test cases into a test case hierarchy (a test scenario see, for example, section 2.5.2 of TET_PG, which describes “Test scenario definitions” that specify which test cases of a test suite are to be executed);

indicating that the one or more test cases in the test case hierarchy are to be executed on the computer program (see, for example, section 5.3.2 of TET_UG);

providing an interface to the test case hierarchy in order to recognize and execute the one or more test cases regardless of the language or format in which the one or more test cases were written (test case controller; see sections 2.1, 2.2, and 2.4 of TET_UG; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG); and

running each of the one or more test cases in the test case hierarchy to test the computer program (test case managers and API libraries; see section 2.4 of TET_UG; see also section 2.4.4 of TET_PG describing the handling of non-API test cases; the test cases are built and executed, regardless of their source language, through the same test case controller; see, for example, the description of build mode in section 6.2.3 of TET_UG).

As per claims 25-27, TETware is further disclosed with executing the one or more test cases on a thread pool comprising one or more threads, and further testing single-threaded and multi-threaded (thread-safe) models (see section 17.4 of TET_PG).

Claim Rejections - 35 USC § 103

5. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

6. Claim 3 is rejected under 35 U.S.C. 103(a) as being unpatentable over TETware and the associated cited documentation as applied to claim 1 above, and further in view of U.S. Patent No. 6,505,342 to Hartmann et al.

As per claim 3, TETware is disclosed with such a system (see disclosure applied above to claim 1), but is not expressly disclosed with a COM technology architecture. However, Hartmann et al. teach a system for testing components that use middleware, such as COM/DCOM (see column 2, line 61 through column 3, line 4). Therefore, it would have been obvious to one having ordinary skill in the computer art at the time the invention was made to modify the system of TETware to include a COM architecture as per the teaching of Hartmann et al. One would be motivated to do so to gain the advantage of supporting and testing implementations in a standardized object-oriented middleware.

Conclusion

7. Any new ground(s) of rejection presented in this Office action were necessitated by Applicant's amendment. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

8. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Eric B. Kiss whose telephone number is (571) 272-3699. The Examiner can normally be reached on Tue. - Fri., 7:00 am - 4:30 pm. The Examiner can also be reached on alternate Mondays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Tuan Dam, can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Art Unit: 2192

Any inquiry of a general nature should be directed to the TC 2100 Group receptionist:

571-272-2100.

EBK / ~~EBK~~
September 30, 2005

A handwritten signature in black ink, appearing to read 'Tuan Dam', with a stylized, flowing script.

TUAN DAM
SUPERVISORY PATENT EXAMINER